



UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/542,189	04/04/2000	Gordon Taylor Davis	RAL9-2000-0008-US1	5890

26675 7590 04/03/2006
DRIGGS, HOGG & FRY CO. L.P.A.
38500 CHARDON ROAD
DEPT. IRA
WILLOUGBY HILLS, OH 44094

EXAMINER

HUISMAN, DAVID J

ART UNIT	PAPER NUMBER
----------	--------------

2183

DATE MAILED: 04/03/2006

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary

Application No.

09/542,189

Applicant(s)

DAVIS ET AL.

Examiner

David J. Huisman

Art Unit

2183

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

- 1) ☒ Responsive to communication(s) filed on 26 January 2006.
- 2a) ☒ This action is **FINAL**. 2b) ☐ This action is non-final.
- 3) ☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

- 4) ☒ Claim(s) 1,3,5-11,13,15-23,27,29-33 and 35 is/are pending in the application.
- 4a) Of the above claim(s) _____ is/are withdrawn from consideration.
- 5) ☐ Claim(s) _____ is/are allowed.
- 6) ☒ Claim(s) 1,3,5-11,13,15-23,27,29-33 and 35 is/are rejected.
- 7) ☐ Claim(s) _____ is/are objected to.
- 8) ☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

- 9) ☐ The specification is objected to by the Examiner.
- 10) ☒ The drawing(s) filed on 04 April 2000 is/are: a) ☒ accepted or b) ☐ objected to by the Examiner.
Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).
Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).
- 11) ☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

- 12) ☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
- a) ☐ All. b) ☐ Some * c) ☐ None of:
1. ☐ Certified copies of the priority documents have been received.
 2. ☐ Certified copies of the priority documents have been received in Application No. _____.
 3. ☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

- | | |
|---|---|
| 1) <input type="checkbox"/> Notice of References Cited (PTO-892) | 4) <input type="checkbox"/> Interview Summary (PTO-413) |
| 2) <input type="checkbox"/> Notice of Draftsperson's Patent Drawing Review (PTO-948) | Paper No(s)/Mail Date. _____ |
| 3) <input type="checkbox"/> Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08) | 5) <input type="checkbox"/> Notice of Informal Patent Application (PTO-152) |
| Paper No(s)/Mail Date _____ | 6) <input type="checkbox"/> Other: _____ |

DETAILED ACTION

1. Claims 1, 3, 5-11, 13, 15-23, 27, 29-33, and 35 have been examined.

Papers Submitted

2. It is hereby acknowledged that the following papers have been received and placed of record in the file: RCE and Amendment as received on 8/10/2005.

Withdrawn Rejections

3. Applicant, by way of amendment, has overcome the prior art rejections set forth in the previous Office Action for claims 1, 3, 5-11, 13, 15-23, and 32-33. Consequently, these rejections are hereby withdrawn by the examiner.

Double Patenting

4. The nonstatutory double patenting rejection is based on a judicially created doctrine grounded in public policy (a policy reflected in the statute) so as to prevent the unjustified or improper timewise extension of the "right to exclude" granted by a patent and to prevent possible harassment by multiple assignees. See *In re Goodman*, 11 F.3d 1046, 29 USPQ2d 2010 (Fed. Cir. 1993); *In re Longi*, 759 F.2d 887, 225 USPQ 645 (Fed. Cir. 1985); *In re Van Ornum*, 686 F.2d 937, 214 USPQ 761 (CCPA 1982); *In re Vogel*, 422 F.2d 438, 164 USPQ 619 (CCPA 1970); and *In re Thorington*, 418 F.2d 528, 163 USPQ 644 (CCPA 1969).

A timely filed terminal disclaimer in compliance with 37 CFR 1.321(c) may be used to overcome an actual or provisional rejection based on a nonstatutory double patenting ground provided the conflicting application or patent is shown to be commonly owned with this application. See 37 CFR 1.130(b).

Effective January 1, 1994, a registered attorney or agent of record may sign a terminal disclaimer. A terminal disclaimer signed by the assignee must fully comply with 37 CFR 3.73(b).

5. Claims 1, 6, 8-10, 11, 16, and 18-20 are rejected under the judicially created doctrine of obviousness-type double patenting as being unpatentable over claims 2, 3, 4-6, 2, 3, and 4-6,

Art Unit: 2183

respectively, of Davis et al., U.S. Patent No. 6,931,641 (herein referred to as Davis) in view of Kruse and Ryba, "Data Structures And Program Design in C++," 1999 (as applied in the previous Office Action and herein referred to as Kruse).

When comparing claim 1 of the instant application to claim 2 of Davis, it can be seen that the differences include:

a) claim 1 of the instant application refers to a "network processor" while claim 2 of Davis refers to a "processor." However, the word "network" in claim 1 is recited in the preamble and does not breath life into the claim. Consequently, the word "network" is given no weight. However, even if "network" were given weight, the examiner asserts that network processors are well known in the art as being processors which efficiently handle applications such as network routing, packet processing, and any other network related functions. As a result, it would have been obvious to modify Davis' processor to be a network processor in order to allow it to efficiently handle network applications.

b) claim 1 of the instant application refers to "accessible data available in a tree search structure" while claim 2 of Davis refers to "accessible data". However, Kruse has taught that search trees are quick and efficient for inserting, deleting, and searching for data. See section 10.2 on pages 444-445 (supplied with a previous Office Action). As a result, it would have been obvious to modify Davis such that the data is accessible via a tree search structure.

c) claim 1 of the instant application refers to stalling "due to a short latency event" and stalling "due to a long latency event" while claim 2 of Davis refers to "a short latency event that causes execution of the first thread to stall for a first predefined time interval" and "a long latency event that causes execution of the first thread to stall for a second predefined time interval." The

Art Unit: 2183

examiner asserts that the differences here are inherent. That is, short and long latency events will inherently have first and second predefined time intervals associated with them (i.e., all stalling takes time).

d) claim 1 of the instant application refers to “when the stall is due to a long latency event, retaining full control by the next thread until the next thread becomes blocked” while claim 2 of Davis refers to “control of the execution is not returned to the first thread when execution of the first thread stalls due to a long latency event.” The examiner asserts that if there are two threads and control is not returned to the first thread, then the option would be to either have a next thread retain control or do nothing at all. As is known in the art, it is much more desirable to do some useful processing as opposed to the system sitting idle. Consequently, it would have been obvious to modify David such that a next thread retains control. And, while claim 2 of Davis does not say how long the control is retained, claim 1 of Davis does say that control is passed to another thread upon the occurrence of a latency event, which inherently blocks a thread.

Consequently, it would’ve been obvious to one of ordinary skill in the art at the time of the invention to modify Davis such that control is retained by the next thread until the next thread is blocked because Davis states that when threads are blocked, control is passed to a next thread.

6. Claims 6 and 8-10 of the instant application are similarly rejected (as above) under the judicially created doctrine of obviousness-type double patenting as being unpatentable over claims 3, and 4-6 of Davis, respectively.

7. Claims 11, 16, and 18-20 of the instant application are similarly rejected (as above) under the judicially created doctrine of obviousness-type double patenting as being unpatentable over claims 2, 3, and 4-6 of Davis, respectively.

Claim Rejections - 35 USC § 103

8. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

(a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negated by the manner in which the invention was made.

9. Claims 27, 29-31, and 35 are rejected under 35 U.S.C. 103(a) as being unpatentable over Cutler et al., U.S. Patent No. 5,752,031 (as applied in the previous Office Action and herein referred to as Cutler), in view of Cota-Robles, U.S. Patent No. 6,658,447 (as applied in the previous Office Action and herein referred to as Cota), and further in view of Nomura et al., U.S. Patent No. 5,327,526 (as applied in the previous Office Action and herein referred to as Nomura).

10. Referring to claim 27, Cutler has taught a thread execution control useful for the efficient execution of independent threads comprising:

a) a priority FIFO buffer for storing thread numbers (see Fig.4-5 and note the ready queue), said buffer including:

a1) means for loading a thread number into the buffer when a packet is dispensed to the processor. See Fig.4 and column 9, lines 63, to column 10, line 3, and note that a request packet is sent to the processor and a thread is initialized and loaded into the ready queue.

a2) means for unloading a thread number from the buffer when a packet has been enqueued for transmission. Clearly, a thread that is loaded into the queue will be removed when it is set to execute. Also, when a thread is finished, it should be removed from the queue.

a3) Cutler has not taught thread number transfer from highest priority to lowest priority in the buffer when the thread with the highest priority encounters a long latency event. However, Cota has taught that threads that have high execution latencies (for instance, accesses to slower processor resources), should be made lower priority threads. See column 7, lines 48-57. As would be realized by one of ordinary skill in the art, an executing thread (the thread currently having the highest priority) that will require a lot of time to execute (due to slower resource access) will also hold up other threads (even those that may execute quickly). Consequently, it would have been obvious to one of ordinary skill in the art at the time of the invention to modify Cutler such that when a long latency event occurs in Cutler, the corresponding thread is assigned lower priority. By doing this, it is ensured that this thread will not hold up higher priority threads.

a4) Cutler has not taught thread outlets of the buffer used to determine priority depending on the length of time a thread has been in the buffer. However, Nomura has taught the general concept of increasing priority of jobs in a queue based on the amount of time spent in the queue. See Fig. 7A-7F and column 6, lines 13-66, and column 7, lines 12-22. As disclosed by Nomura, such a scheme, which is known in the art, allows an initially low priority job to eventually be executed even though initially higher priority jobs may be queued. This essentially prevents the low priority job from never being executed if all if nothing but higher priority jobs are subsequently queued. This scheme also applies to threads in that if there is always a higher priority thread in the queue, a low priority thread would never execute. Nomura's scheme, however, ensures that over time, low priority threads would be executed. As a result, it would

have been obvious to one of ordinary skill in the art at the time of the invention to modify Cutler to include the priority-increasing scheme of Nomura.

b) Cutler has further taught a plurality of thread control state machines for use in shifting execution control between threads upon occurrence of latency events. See Fig.4, and note that each thread's state must be tracked (i.e., it must be tracked whether each thread is in the ready state, standby state, running state, waiting state, etc., so that the next state for the thread may be determined). Also, it is clear that these state machines are used in determining execution control.

c) Cutler in view of Cota and further in view of Nomura has taught an arbiter for determining the thread execution priority among multiple threads based upon signals outputted from the FIFO buffer and the state machines. It is inherent that if one of a plurality of threads is selected, then some component, more specifically, an arbiter, must determine which thread to select. The highest-priority thread that is awaiting processing will be selected for execution. See Fig.4-5 of Cutler and also recall the discussion of Cota and Nomura. This selection is based on signals from the state machines (signals specifying what state the threads are in...the only threads that may be selected for dispatch/execution are those in the ready/standby state) and signals from the FIFO (priorities).

11. Referring to claim 29, Cutler in view of Cota and further in view of Nomura has taught a thread execution control as described in claim 27. Cutler in view of Cota and further in view of Nomura has further taught that the arbiter controls the priority of execution of multiple independent threads based on the Boolean expression recited in claim 29 comprising:

a) determining whether a request R is active or inactive. See Fig.4 of Cutler and note that only threads in the ready/standby state can make active requests for dispatch/execution. Threads

Art Unit: 2183

being initialized, or terminated, or those that are already executing, do not need to make requests for execution. Clearly, each thread is associated with a request, but for those that have already had their request satisfied, and are executing, they're requests are inactive (i.e., they're not making requests anymore).

b) determining the priority of the threads. See Cutler, Fig.4, and note that threads are preempted by higher priority threads. Also recall the discussion of Cota and Nomura above.

c) matching the request R with the corresponding thread P. Clearly, only requesting (ready) threads may be executed. Of the requesters, only the highest priority will be chosen. See Fig.4-5 of Cutler, and also recall the priority schemes of Cota and Nomura.

d) granting a request for execution if the request is active and if the corresponding thread P has the highest priority. If a thread is ready to execute and therefore requesting execution, if it is the higher priority thread, then it will be granted execution. See Fig.4-5 of Cutler, and recall Cota and Nomura, which talk about priority of items in a queue.

It should be noted that the formula of claim 29 represents the algorithm set forth in steps (a)-(d) of claim 29, and Joy has taught steps (a)-(d), then Joy has also taught the formula even though it is not explicitly stated.

12. Referring to claim 30, Cutler in view of Cota and further in view of Nomura has taught thread execution control as described in claim 27. Cutler has taught control logic to:

a) dispatch a packet to a thread. See column 9, lines 63-66, and note that a request packet is received and a thread is initialized to handle the request.

b) move the thread from the initialize state to a ready state. See Fig.4, and note the path from 40a to 40b.

c) request execution cycles for the thread. Clearly, by being in the ready state, the threads are requesting execution.

d) move the thread to the execute state upon grant by the arbiter of an execution cycle. See Fig.4 and note that an arbiter will choose to either preempt a thread standing by for execution or send a thread to an execute stage 40d.

e) continue to request execution cycles while the thread is queued in the execute state. Again, the thread will continue to request execution cycles until it completes or some other event happens (preemption or lack of resources).

f) return the thread to the initialize state if there is no latency event, or send the thread to the wait state upon occurrence of a latency event. See Fig.4.

13. Referring to claim 31, Cutler in view of Cota and further in view of Nomura has taught thread execution control as described in claim 27. Cutler has further taught that the FIFO buffer further includes control logic to detect occurrence of latency events. See the path from 40d to 40e, and note that a thread must wait for kernel objects. This waiting is a latency event. Cutler has not taught and differentiating between a short latency event and a long latency event.

However Official Notice that the implementation of multiple levels of cache and main memory are well known and accepted in the art. More specifically, memory allows for storage of massive amounts of data at a relatively cheap price while caches allow for storage and fast retrieval of the most recently accessed data. As a result, in order to allow for efficient storage of data it would have been obvious to implement caches and memories in Cutler. In addition, systems having caches and main memory also inherently detect cache misses and main memory accesses. For instance, in a system with a first level (L1) cache, a second level (L2) cache, and a main

Art Unit: 2183

memory, the system will detect a short latency event (miss at the L1 cache), and the system will be able to detect a long latency event (miss at the L2 cache, which results in a main memory access).

14. Claim 35 is rejected under 35 U.S.C. 103(a) as being unpatentable over Cutler in view of Cota and further in view of Nomura, as applied above, in view of Parady, U.S. Patent No. 5,933,627 (as applied in the previous Office Action) and further in view of Flynn et al., U.S. Patent No. 6,052,708 (as applied in the previous Office Action and herein referred to as Flynn). Referring to claim 35, Cutler in view of Cota and further in view of Nomura has taught a thread execution control as described in claim 27. Cutler in view of Cota and further in view of Nomura has not taught a separate instruction buffer for each execution thread. However, Parady has taught such a concept. See Fig.3 and column 5, lines 6-10. Parady has further disclosed that upon a thread switch, the stream of instructions in one of the instruction buffers will simply pick up where it left off. See column 3, lines 51-56. This would eliminate having to flush a single instruction buffer (if only one buffer were used to store instructions for all of the threads) and refilling it with the correct thread's instructions. Therefore, it would have been obvious to one of ordinary skill in the art at the time of the invention to provide a prefetch buffer for each thread in Cutler et al, in order to provide efficient switching. Parady has not explicitly taught collecting instructions in a prefetch buffer for its execution thread. However, Flynn has taught such a concept. See column 4, lines 43-53. Note that when a thread is inactive, a thread's instructions may be fetched so that when the thread is made active, the instruction will be ready for dispatch immediately, thereby increasing efficiency. As a result, it would have been obvious to one of

Art Unit: 2183

ordinary skill in the art at the time of the invention to modify Cutler et al in view of Parady such that each buffer taught by Parady is a prefetch buffer and instructions for an inactive thread are prefetched into its prefetch buffer.

Response to Arguments

15. On page 11 of the remarks (1st paragraph), applicant states that a partial or complete terminal disclaimer will be filed upon allowance of one or more of the claims rejected under double patenting. The examiner asserts that the claims cannot be allowed until the double-patenting is overcome.

Conclusion

16. Applicant's amendment necessitated the new ground(s) of rejection presented in this Office action. Accordingly, **THIS ACTION IS MADE FINAL**. See MPEP § 706.07(a). Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire **THREE MONTHS** from the mailing date of this action. In the event a first reply is filed within **TWO MONTHS** of the mailing date of this final action and the advisory action is not mailed until after the end of the **THREE-MONTH** shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than **SIX MONTHS** from the date of this final action.

Art Unit: 2183

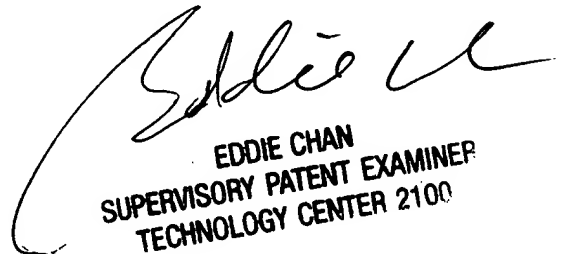
Any inquiry concerning this communication or earlier communications from the examiner should be directed to David J. Huisman whose telephone number is (571) 272-4168.

The examiner can normally be reached on Monday-Friday (8:00-4:30).

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Eddie Chan can be reached on (571) 272-4162. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see <http://pair-direct.uspto.gov>. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

DJH
David J. Huisman
March 17, 2006



EDDIE CHAN
SUPERVISORY PATENT EXAMINER
TECHNOLOGY CENTER 2100